Original Article
International Journal of Fuzzy Logic and Intelligent Systems
Vol. 18, No. 3, September 2018, pp. 196-203
http://doi.org/10.5391/IJFIS.2018.18.3.196

ISSN(Print) 1598-2645
ISSN(Online) 2093-744X

Semantic Segmentation Using Trade-Off and Internal Ensemble

Wang-Su Jeon[1], Grzegorz Cielniak[2], and Sang-Yong Rhee[3]
[1]Department of IT Convergence Engineering, Kyungnam University, Changwon, Korea
[2]School Computer Science, University of Lincoln, Lincoln, UK
[3]Department of Computer Engineering, Kyungnam University, Changwon, Korea

ljfis

Abstract

The computer vision consists of image classification, image segmentation, object detection, and tracking, etc. Among them, image segmentation is the most basic technique of the computer vision, which divides an image into foreground and background. This paper proposes an ensemble model using a concept of physical perception for image segmentation. Practically two connected models, the DeepLab and a modified VGG model, get feedback each other in the training process. On inference processing, we combine the results of two parallel models and execute an atrous spatial pyramid pooling (ASPP) and post-processing by using conditional random field (CRF). The proposed model shows better performance than the DeepLab in local area and about 1% improvement on average on comparison of pixel-by-pixel.

Keywords: Convolution neural network, Correlation, Internal ensembles semantic segmentation, Conditional random field

1. Introduction

The field of artificial intelligence has evolved due to improved performance of hardware and big data technology. A lot of studies to imitate the human visual recognition process in which the image is transmitted from the retina to the visual cortex has been carried in the computer vision corresponding to the visual part of human among the various artificial intelligence technologies. Computer vision enables a robot to visualize and analyze any scene. The basic techniques for image analysis are image classification, image segmentation, object detection, and location tracking. Among them, image segmentation is one of the most basic method for the image analysis which divides the image into regions that have the same characteristics. After preprocessing processes are performed to remove noise, a method of expanding a region by combining pixels having similar characteristics by using local homogeneity, or separating a region into two or more ones according to the different characteristics has been executed in the conventional image segmentation techniques. However, it was very hard to make joining/splitting rules because there exist multifarious relationships between neighboring regions [1–6]. To solve this problem, methods of generating rules through learning mechanisms have been suggested. A representative method is a deep neural network, which was initially applied to object recognition.

The good performance of deep neural networks was shown at the ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2012. Studies using classification rules by researchers had been a mainstream by 2011, with the best top 5 recognition rate at 74.2%.

Received: Jul. 21, 2018
Revised : Sep. 6, 2018
Accepted: Sep. 18, 2018

Correspondence to: Sang-Yong Rhee
(syrhee@kyungnam.ac.kr)

However, AlexNet [7], designed by Supervision team of University of Toronto in 2012 based on the convolutional neural network (CNN), shows outstanding performance. It was superior to methods using features specified by researchers or machine learning methods that classify data after preprocessing. The recognition rate of the CNN started to rise sharply from 83.6% and reached 97.749% in 2017.

Hereafter CNN, which was mainly used in the field of image recognition, has been used in the other areas such as image classification, object detection, and semantic image segmentation. In 2014, Berkeley University showed that it could use the CNN for semantic segmentation and was added skip-connection called unpooling to ImageNet models AlexNet, ZFNet [8], VGGNet [9] and GoogleNet [10] and fully convolutional networks (FNC) [11]. After that, models such as DeconvNet [12] and SegNet [13] have been developed, and the Google research team proposed a broader atrous convolution method, reducing the computational complexity of the convolution. The developed model is DeepLab [14, 15], based on VGGNet and ResNet [16]. Since then, the research direction of semantic image segmentation has been focused on real-time processing.

Human perception is consists of the constructive perception and the direct perception. The directive perception is a concept that sensation allows us to actually perceive an object. Constructive perception, on the other hand, refers to performing cognitive comprehension of stimuli by using information from the visual sensor and other sources of information, such as memory, on viewing a new object. The CNN belongs to the directive perception. In this paper, we pursue the constructive perception.

Because existing neural networks could extract various features according to their model design, they must have complex or deep structures such as GoogleNet and ResNet. The ensemble model has also been studied to unify the shallow or deep models into a single model [17]. The ensemble learning method has a disadvantage that it takes a long time because it has to be repeatedly executed and have to do learning procedure by an end-to-end method.

In this paper, we propose a model that has two CNN information flow lines. The lines connected to each other and have a mutual relationship to learn and get feedback information in backward propagation learning process. In case of the existing ensemble model, inference is performed by combining the learning results of each model, but the proposed model changes the ensemble model to be able to establish a mutual relationship within the model. This method can be regarded as a combination of the existing end-to-end learning and the ensemble method.

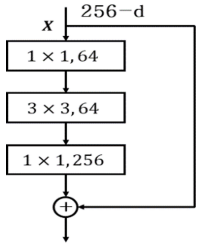


Figure 1. Residual bottleneck architecture.

The rest of paper is organized as follows: Section 2 describes the proposed system's architecture. In Section 3, experimental results and analysis of the proposed method are described. Finally, Section 4 describes the experimental results and future research directions.

2. The Proposed System

2.1 The Overview

Most of existing neural networks infer results through a single model. If two models are used, different features can be learned, and the training process can have mutual relationship in which information can be exchanged and learned. To design the interrelated model structure, we use the ResNet structure of the DeepLab as a basic structure.

When we use the existing VGGNet and GoogleNet, there is a problem that the model did not learn well with an image of over 30 layers. To solve this problem, ResNet introduces a bottleneck structure as shown in Figure 1, which can increase the network depth to 50-150 layers. The bottleneck structure can reduce the amount of computation and control the dimension by sequentially using 1×1 convolution, 3×3 convolution, and 1×1 convolution and a detour path. The detour path is to add the previous value which can solve the gradient vanishing problem that happens when the network is deepened by using the residual connection.

DeepLab uses the ResNet-100 Layer structure as shown in Figure 2. fifty out of the 100 layers use the existing convolution operation, and the remaining 50 layers use the atrous convolution. If the 3×3 structure extracting feature in the bottleneck structure of Figure 1 is replaced by the atrous convolution, it shows as Figure 2.

The inference process of the DeepLab using atrous convolution is as follows: When an image is input as shown in Figure

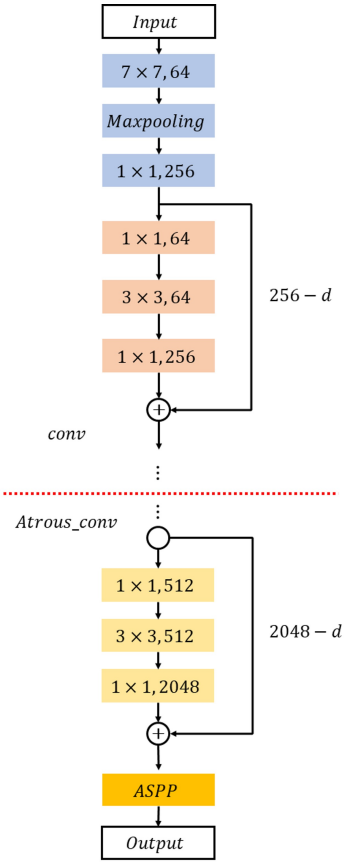


Figure 2. ResNet architecture with atrous convolution.

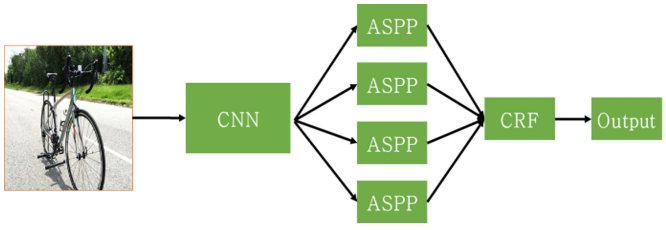


Figure 3. The base model of the inference process.

3, the CNN extracts the features. After the image size is adjusted by using ASPP layers that imitated the image pyramid technique at the last stage, the results are combined. The final inference is generated by post-processing using the CRF (Conditional Random Field).

2.2 The Proposed System Structure

In the existing ensemble methods, several models are constructed in parallel, trained and combined the results on inferring. In this paper, two models are connected in parallel, but the two models are mutually related to each other to exchange information in training process. We use the DeepLab as a central structure and additionally use one more model which is a modified VGGNet.

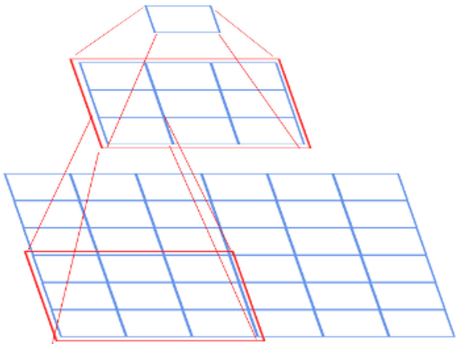
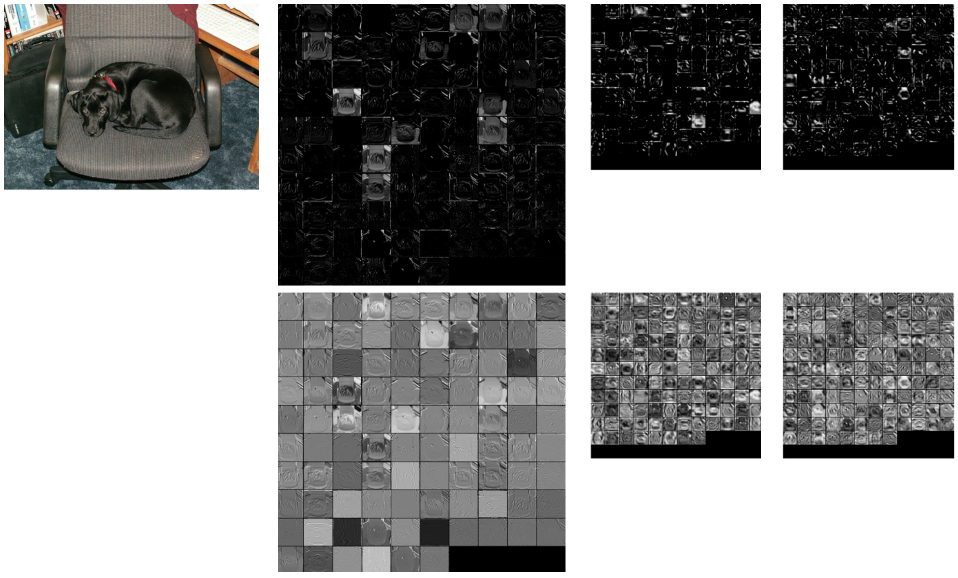


Figure 4. Factorizing convolution architecture.

The factorizing convolution of the VGGNet is used as much as possible in designing this model. It improved performance by minimizing the number of parameters by stacking two or three layers of 3×3 filters rather than one layer of 5×5 or 7×7 filters. Figure 4 shows the factorizing convolution. Figure 5 is what the convolution operation is visualized by using a dog image lying on a chair. It shows the intermediate process extracted through two pipelines. In Figure 5(b), the overall outline of the image is extracted roughly, and Figure 5(c) and (d) show that features are extracted in detail on continuing the process.

As shown in Figures 4 and 5, when 3×3 filters are stacked on three layers, its computation amount is smaller than 7×7 filter and the reception range is the same. This characteristic is applied to the proposed model. The structure of the existing VGGNet is a model that increases the reception range by stacking two layers of 3×3 layers in the front, and then stacking three layers in the rear part. However, in the case of the proposed model, the 3×3 filters are stacked on three layers from the first stage. It makes wide acceptance range and the features are extracted by using them. That is why it is necessary to adjust the size to 40×40 to exchange information mutually with the existing DeepLab structure. In order to adjust the size, the fourth pooling operation of the existing VGGNet structure has to be removed. In this case, the acceptance range became insufficient. So we modify the structure as Figure 6 to solve the problem.

Figure 6 shows the structure of the additional model. Its detailed structure is shown in Table 1. The conv_module in Table 1 is an acronym that combines the process of convolution, batch norm, and Relu. The proposed model computes three times using the 3×3 conv_module as shown in Table 2 when



(a) (b) (c) (d)

Figure 5. Visualization of convolution computing: (a) input, (b) conv1, (c) conv2, (d) conv3.

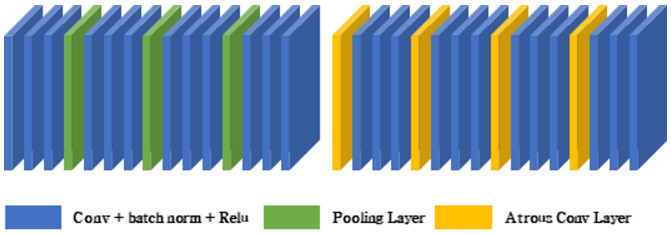


Figure 6. Architecture of the modified model.

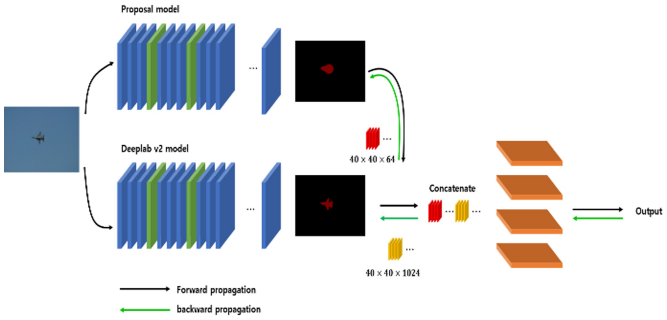


Figure 7. The training process of the proposed architecture.

an image of 321×321 size is given, and then adjusts its size by max pooling. The atrous convolutions are performed at its back part. First two operations execute as two rate each and the next two ones as four rate each. Eventually the model has an output of $40 \times 40 \times 64$, and the feature maps are linked to the DeepLab model, and the two models interactively are trained.

The training process of the proposed one is shown in Figure 7. When an image is the input, the proposed model produces complementary output by connecting the feature maps of the two models. In the backward process, the two models are mutually trained.

The inference process is shown in Figure 8. An image is given to the two models simultaneously. After the feature maps of the two models are combined, they pass through the ASPP and then post-processed with the CRF. Figure 9 shows alteration before and after CRF post-processing.

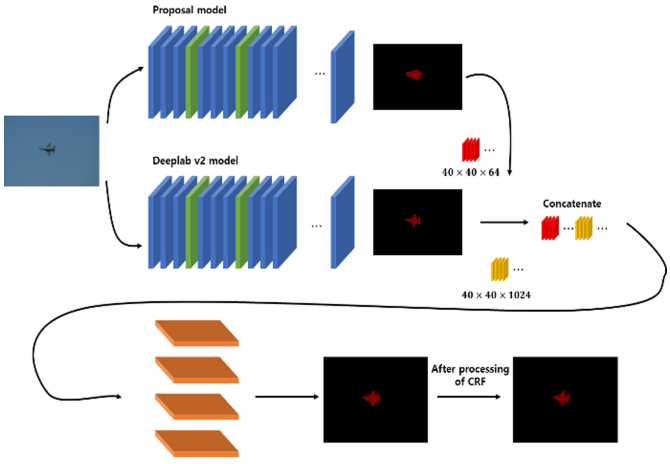


Figure 8. The inference process of the proposed architecture.

Table 1. Detailed configuration of the added architecture

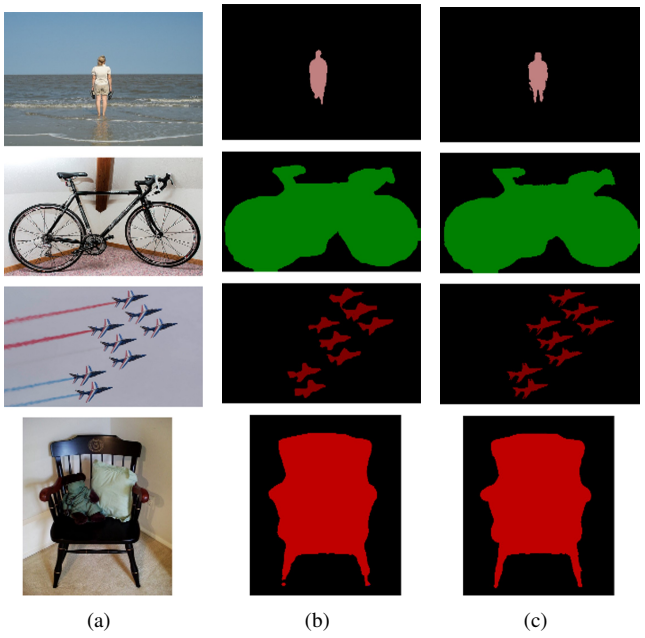
Type	Size	Stride	Rate	Input size
Conv_module	3×3	1	-	$321 \times 321 \times 64$
Conv_module	3×3	1	-	$321 \times 321 \times 64$
Conv_module	3×3	1	-	$321 \times 321 \times 64$
Max Pooling	2×2	2	-	$160 \times 160 \times 64$
Conv_module	3×3	1	-	$160 \times 160 \times 128$
Conv_module	3×3	1	-	$160 \times 160 \times 128$
Conv_module	3×3	1	-	$160 \times 160 \times 128$
Max Pooling	2×2	2	-	$80 \times 80 \times 128$
Conv_module	3×3	1	-	$80 \times 80 \times 256$
Conv_module	3×3	1	-	$80 \times 80 \times 256$
Conv_module	3×3	1	-	$80 \times 80 \times 256$
Max Pooling	2×2	2	-	$40 \times 40 \times 512$
Conv_module	3×3	1	-	$40 \times 40 \times 512$
Conv_module	3×3	1	-	$40 \times 40 \times 512$
Conv_module	3×3	1	-	$40 \times 40 \times 512$
Atrous Conv	3×3	1	2	$40 \times 40 \times 512$
Conv_module	3×3	1	-	$40 \times 40 \times 512$
Conv_module	3×3	1	-	$40 \times 40 \times 512$
Conv_module	3×3	1	-	$40 \times 40 \times 512$
Atrous Conv	3×3	1	2	$40 \times 40 \times 256$
Conv_module	3×3	1	-	$40 \times 40 \times 256$
Conv_module	3×3	1	-	$40 \times 40 \times 256$
Conv_module	3×3	1	-	$40 \times 40 \times 256$
Atrous Conv	3×3	1	4	$40 \times 40 \times 128$
Conv_module	3×3	1	-	$40 \times 40 \times 128$
Conv_module	3×3	1	-	$40 \times 40 \times 128$
Conv_module	3×3	1	-	$40 \times 40 \times 128$
Atrous Conv	3×3	1	4	$40 \times 40 \times 64$
Conv_module	3×3	1	-	$40 \times 40 \times 64$
Conv_module	3×3	1	-	$40 \times 40 \times 64$
Conv_module	3×3	1	-	$40 \times 40 \times 64$

Conv_module: Conv, batch norm, Relu

3. Experiments and the Results

3.1 Experimental Environment

The experimental environment was as follows: The operating system was Linux Ubuntu 16.04.2 64 bit. The hardware, consisted of an Intel i7-6770k CPU, 32 GB memory, two NVIDIA TITAN X, and Tensorflow 1.2.1 is used by deep learning framework. The PASCAL VOC (PASCAL Visual Object Classes Challenge) 2012 [18, 19] dataset was used. It consists of 1,464 training data and 1,456 test data. We used 1,000 epoch, 4 batch size, and the learning rate 0.0001 as hyper parameters, and momentum [20] is used as a stochastic gradient descent method.



(a) (b) (c)

Figure 9. Alteration after post-processing: (a) input, (b) before CRF, (c) after CRF.

In this paper, we compared the DeepLab v1 and v2 for performance evaluation of the proposed method. The images used in the experiment were adjusted to the 500×375 and 321×321 sizes, respectively.

3.2 Results

The three models, which were the proposed model, the DeepLab v1 and DeepLab v2, were tested using eight images. The first line in Figure 10 shows segmentation results. The proposed model segmented better for the handles of the bicycle and the legs of the person, but the DeepLab v2 is better at the fork of the front wheel. The DeepLab v2 has a slightly wider area, which is better at the second line.

The proposed one was better because the results were more detailed from the third to the fifth line. In the sixth line, the outline of a table and the shape of a water bottle from proposed model were most similar to the ground truth. The front wheel part of a plane well was segmented in seventh line and outlines of two motorcycles are more similar to the ground truth in the last line. In conclusion the proposed method has the effect of correcting the position of objects and can represent even details. Table 2 [21] shows that the proposed model performed better using numerical values.

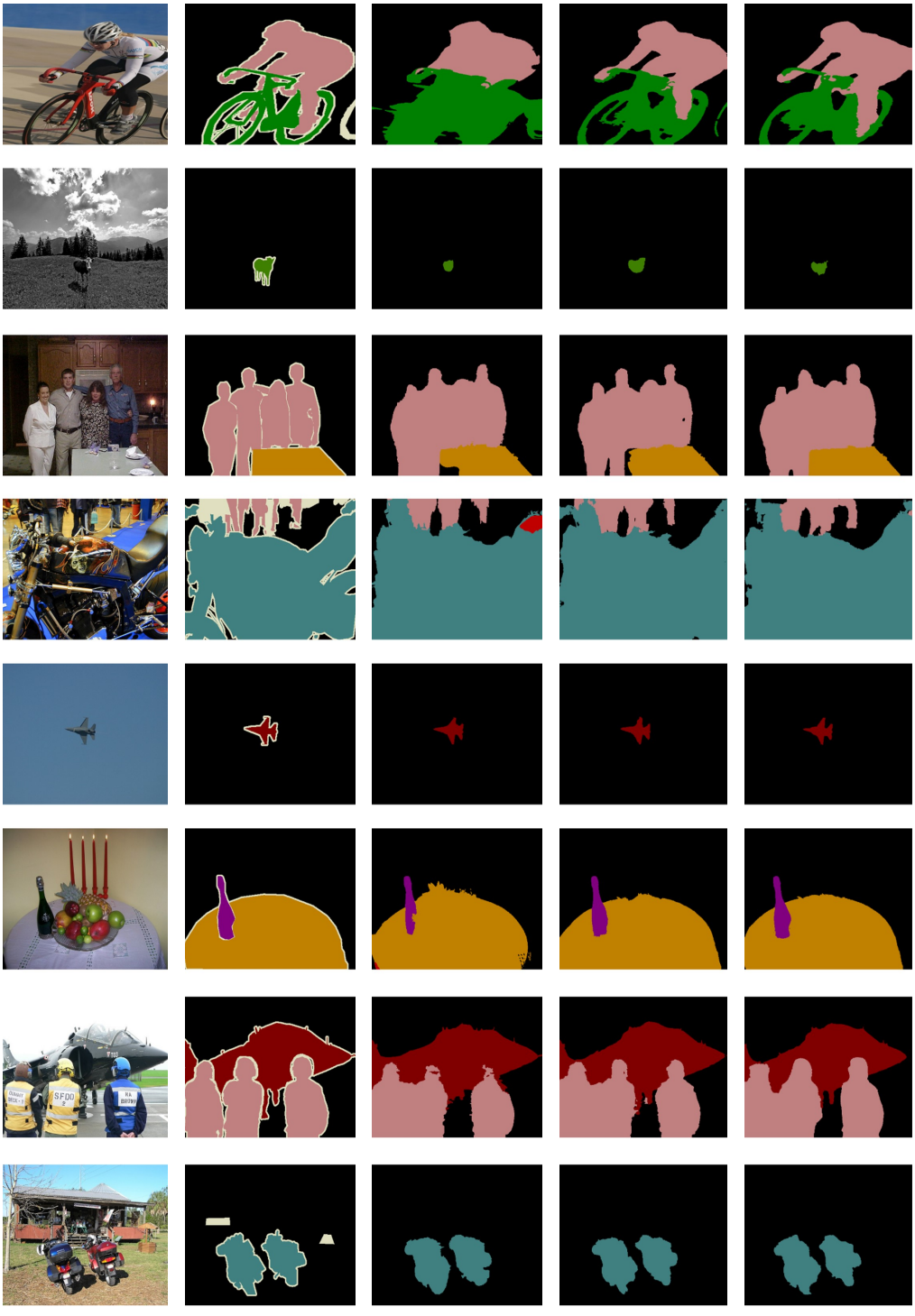


Figure 10. Compare results by model: (a) input, (b) ground truth, (c) DeepLab v1, (d) DeepLab v2, (e) Ours.

4. Conclusions

In this paper, we proposed the model that connects two models and performs complementary operations to perform semantic segmentation. A DeepLab model and modified VGGNet model were placed in parallel and information was exchanged with each other. The DeepLab v1, v2 and the proposed model were trained using the PASCAL VOC 2012 dataset and evaluated.

Table 2. Accuracy rate (%) by model

Category	DeepLab v1	DeepLab v2	Ours
background	92.1	93.2	96.6
aeroplane	78.4	92.6	94.6
bicycle	33.1	60.4	63.6
bird	78.2	91.6	92.9
boat	55.6	63.4	64.4
bottle	65.3	96.3	96.1
bus	81.3	95.0	95.6
car	75.5	88.4	89.7
cat	78.6	82.6	83.5
chair	25.3	32.7	31.4
cow	69.2	88.5	90.2
dining table	52.7	67.6	67.2
dog	75.2	89.6	91.1
horse	69.0	92.1	93.2
motorbike	79.1	87.0	86.5
person	77.6	87.4	88.2
pottle plant	54.7	63.3	65.1
sheep	78.3	88.3	87.4
sofa	45.1	60.0	59.1
train	73.3	86.8	86.3
TV/monitor	56.2	74.5	75.8
Mean IOU	66.4	79.7	80.5

We could get the results that the performance was improved by from 1% to 3% on comparing as the pixel level, and by about 1% on the average.

However, the proposed model needed long training time, and inference process took 1.5 times as long. In the future, we will carry out the research to improve the inference speed and to reduce the training time by modifying the structure.

Conflict of Interest

No potential conflict of interest relevant to this article was reported.

Acknowledgements

This paper is an excerpt from a part of an Master's thesis.

References

[1] J. W. Ko, B. I. Choi, and F. C. H. Rhee, "A density estimation based fuzzy C-means algorithm for image segmentation," *Journal of Korean Institute of Intelligent Systems*, vol. 17, no. 2, pp. 196-201, 2007. http://doi.org/10.5391/JKIIS.2007.17.2.196

[2] M. J. Lee, T. S. Jin, and G. H. Hwang, "A study on image segmentation and tracking based on fuzzy method," *Journal of Korean Institute of Intelligent Systems*, vol. 17, no. 3, pp. 368-373, 2007. http://doi.org/10.5391/JKIIS.2007.17.3.368

[3] H. W. Lee, N. R. Kim, and J. H. Lee, "Deep neural network self-training based on unsupervised learning and dropout," *International Journal of Fuzzy Logic and Intelligent System*, vol. 17, no. 1, pp. 1-9, 2017. https://doi.org/10.5391/IJFIS.2017.17.1.1

[4] W. S. Jeon and S. Y. Rhee, "Fingerprint pattern classification using convolution neural network," *International Journal of Fuzzy Logic and Intelligent System*, vol. 17, no. 3, pp. 170-176, 2017. http://dx.doi.org/10.5391/IJFIS.2017.17.3.170

[5] A. Palvanov and Y. I. Cho, "Comparisons of deep learning algorithms for MNIST in real-time environment," *International Journal of Fuzzy Logic and Intelligent System*, vol. 18, no. 2, pp. 126-134, 2018. http://doi.org/10.5391/IJFIS.2018.18.2.126

[6] Y. R. Pandeya and J. W. Lee, "Domestic cat sound classification using transfer learning," *International Journal of Fuzzy Logic and Intelligent System*, vol. 18, no. 2, pp. 154-160, 2018. http://doi.org/10.5391/IJFIS.2018.18.2.154

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing System*, vol. 25, pp. 1097-1105, 2012.

[8] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*. Cham: Springer, 2014, pp. 818-833. https://doi.org/10.1007/978-3-319-10590-1_53

[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large scale image recognition," 2014, Available: https://arxiv.org/abs/1409.1556

[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, 2015, pp. 1-9.

[11] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, 2015, pp. 3431-3440.

[12] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, Chile, pp. 1520-1528. https://doi.org/10.1109/ICCV.2015.178

[13] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: a deep convolutional encoder-decoder architecture for image segmentation," 2016, Available: https://arxiv.org/abs/1511.00561

[14] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," 2016, Available: https://arxiv.org/abs/1412.7062

[15] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "DeepLab: semantic image segmentation with deep convolution nets, Atrous convolution, and fully connected CRFs," 2017, Available: https://arxiv.org/abs/1606.00915

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 2016, pp. 770-778. https://doi.org/10.1109/CVPR.2016.90

[17] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, Available: https://arxiv.org/abs/1503.02531

[18] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: a retrospective," *International Journal of Computer Vision*, vol.111, no. 1, pp. 98-136, 2015. https://doi.org/10.1007/s11263-014-0733-5

[19] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *European Conference on Computer Vision*. Cham: Springer, 2014, pp. 297-312. https://doi.org/10.1007/978-3-319-10584-0_20

[20] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of International Conference on Machine Learning*, Atlanta, GA, 2013, pp. 1139-1147.

[21] PASCAL VOC Challenge performance evaluation and download server, "Segmentation Result: VOC 2012," Available: http://www.iub.edu.bd/files/articles/PASCALVOCChallenge.pdf

Wang-Su Jeon received his B.S. and M.S. degrees in Computer Engineering and IT Convergence Engineering from Kyungnam University, Masan, Korea, in 2016 and 2018, respectively, and is currently pursuing the Ph.D. degree in IT Convergence Engineering at Kyungnam University, Masan, Korea. His present interests include computer vision, pattern recognition and machine learning
E-mail: jws2218@naver.com

Grzegorz Cielniak is a Senior Lecturer at the School of Computer Science, University of Lincoln. He received his M.Sc. in Robotics from the Wrocław University of Technology in 2000 and a Ph.D. in Computer Science from the Örebro University in 2006. His research interests include mobile robotics, artificial intelligence, real-time computer vision systems and multi-sensor fusion with particular focus on robotic applications in agriculture.
E-mail: Grzegorz.Cielniak@Gmail.com

Sang-Yong Rhee received his B.S. and M.S. degrees in Industrial Engineering from Korea University, Seoul, Korea, in 1982 and 1984, respectively, and his Ph.D. degree in Industrial Engineering at Pohang University, Pohang, Korea. He is currently a professor at the Computer Engineering, Kyungnam University, Masan, Korea. His research interests include computer vision, augmented reality, neuro-fuzzy and human-robot interface.
E-mail: syrhee@kyungnam.ac.kr